CLAIMS

What is claimed is:

1. A method comprising:

placing an executable thread of instructions in an inactive state in response to detection of at least one of a set of predetermined conditions;

sending a message from a semaphore to control circuitry to execute the thread of instructions to change a state of the thread of instructions from the inactive state.

2. The method of claim 1 wherein changing the state of the thread of instructions from the inactive state comprises changing the state of the thread of instructions to an active state.

3. The method of claim 2 further comprising executing the thread of instructions when in the active state.

4. The method of claim 1 wherein the set of predetermined conditions comprises an unresolved dependency.

5. The method of claim 1 wherein the set of predetermined conditions comprises a response from the semaphore indicating that a resource corresponding to the semaphore is unavailable.

6. The method of claim 1 further comprising maintaining an indication of a state of each of a plurality of executable threads of instructions.

7. The method of claim 6 wherein the indication of the state of each thread comprises a state variable corresponding to a dependency, if any, of an associated thread.

8. An apparatus comprising:

execution means for placing an executable thread of instructions in an inactive state in response to detection of at least one of a set of predetermined conditions;

communication means communicatively coupled with the execution means for sending a message from a semaphore to control circuitry to execute the thread of instructions to change a state of the thread of instructions from the inactive state.

9. The apparatus of claim 8 further comprising means for maintaining an indication of a state of each of a plurality of executable threads of instructions.

10. The apparatus of claim 9 wherein the indication of the state of each thread comprises a state variable corresponding to a dependency, if any, of an associated thread.

11. An apparatus comprising:

an execution circuit to receive and execute a thread of instructions, wherein the execution circuit transmits a semaphore request message and places the thread in an inactive state in response to the thread of instructions requiring a resource having an associated semaphore; and

a semaphore entity coupled with the execution circuit to receive the semaphore request message from the execution circuit and to selectively grant control of the semaphore in response to the semaphore request message by transmitting a semaphore acknowledge message to the execution circuitry, wherein the execution circuitry, in response to receiving the semaphore acknowledge message, removes the thread of instructions from the inactive state.

12. The apparatus of claim 11 further comprising:

at least one additional execution circuit to execute threads of instructions; and

a thread dispatcher coupled with the execution circuit and at least one additional execution circuit to dispatch threads for execution by selected execution circuits.

13. The apparatus of claim 11, wherein the execution circuitry, in response to receiving the semaphore acknowledge message, resumes execution of the thread of instructions including accessing the resource associated with the semaphore.

14. The apparatus of claim 11 wherein when the thread of instructions is in the inactive state, execution of the instructions ceases and the execution circuitry does not poll the semaphore entity to determine a status of the semaphore request message.

15. An system comprising:

a memory controller;

an execution circuit coupled with the memory controller to receive and execute a thread of instructions, wherein the execution circuit transmits a semaphore request message and places the thread in an inactive state in response to the thread of instructions requiring a resource having an associated semaphore;

a semaphore entity coupled with the execution circuit to receive the semaphore request message from the execution circuit and to selectively grant control of the semaphore in response to the semaphore request message by transmitting a semaphore acknowledge message to the execution circuitry, wherein the execution circuitry, in response to receiving the semaphore acknowledge message, removes the thread of instructions from the inactive state.

16. The system of claim 15 further comprising:

at least one additional execution circuit to execute threads of instructions; and

a thread dispatcher coupled with the execution circuit and at least one additional execution circuit to dispatch threads for execution by selected execution circuits.

17. The system of claim 15, wherein the execution circuitry, in response to receiving the semaphore acknowledge message, resumes execution of the thread of instructions including accessing the resource associated with the semaphore.

18. The system of claim 15 wherein when the thread of instructions is in the inactive state, execution of the instructions ceases and the execution circuitry does not poll the semaphore entity to determine a status of the semaphore request message.